**Step (1)**

* نبود Attribute ای مانند [Microsoft.AspNetCore.Mvc.HttpGet]

زمانی که Action ای را به این صورت می‌نویسیم، با خطای Swagger مواجه می‌شویم، ولی باید دقت داشته باشیم که این مدل نگارش، کاملا کار می‌کند و اتفاقا برای همه پانزده Verb به جز Head عمل می‌کند. این مدل نگارش در حالت Restful به عنوان یک Bad Practice در نظر گرفته می‌شود!

**Step (2)**

زمانی که Action ای را به این صورت می‌نویسیم، این Action صرفا به صورت Get قابل استفاده بوده و در Swagger نیز مشکلی وجود ندارد. فقط مشکل این است که نمی‌توانیم تحت شرایطی خاص، خطایی (Problem) را Return نماییم.

نمونه: Step (3)

**Step (4)**

زمانی که Action ای را به این صورت می‌نویسیم، این Action صرفا به صورت Get قابل استفاده بوده و در Swagger نیز مشکلی وجود ندارد. فقط مشکل این است که در محیط Swagger مشخص نمی‌شود که دقیقا چه خروجی از این Action قابل دریافت می‌باشد.

**Step (5)**

در این حالت نیز، هر چند که IActionResult را به ActionResult تغییر داده‌ایم، ولی کماکان همان مشکل Step (4) وجود دارد.

**Step (6)**

زمانی که Action ای را به این صورت می‌نویسیم، مشکل موجود در Step (4) و Step (5) حل می‌شود و Swagger اعلام می‌کند که در شرایط OK یعنی کد 200، فهرستی از Customer ها برمی‌گردد، ولی مشکلی که وجود دارد آن است که Swagger شناسایی نمی‌کند که این Action، یک خطای 500 نیز ممکن است برگرداند!

**Step (7)**

در این مرحله، از Attribute‌ ای به نام ProducesResponseType می‌کنیم. این عمل را بدین منظور انجام می‌دهیم که به Swagger بفهمانیم که این Action، خروجی Problem نیز می‌تواند داشته باشد. ولی مشکلی که حالا به وجود می‌آید آن است که Swagger دیگر شناسایی نمی‌کند که این Action می‌تواند مجموعه‌ای از Customer ها در در شرایط OK برگرداند!

**Step (8)**

در این مرحله، از ProducesResponseType مجددا استفاده می‌کنیم و اعلام می‌کنیم که این Action می‌تواند علاوه بر Problem، مجموعه‌ای از Customer ها را نیز برگرداند. به همین دلیل دیگر نیازی نیست که از ActionResult به صورت Generic استفاده نماییم، پس صرفا از IActionResult استفاده می‌کنیم.

**Step (9)**

در این مرحله، سعی می‌کنیم که Action خود را به صورت Async بنویسیم. باید دقت داشته باشیم که از نظر نگارش استاندارد، بهتر است که نام Action، به کلمه Async خاتمه پیدا کند.

**Step (10)**

در این مرحله، دو Attribute‌ دیگر به نام‌های Consumes و Produces می‌نویسیم که بدین وسیله اعلام نماییم که دقیقا چه MediaType هایی را این Action، دریافت می‌کند و برمی‌گرداند.

**Step (11)**

در این مرحله، از نظر نگارشی از دستور Region استفاده می‌کنیم و به متن داخل EndRegion نیز توجه می‌کنیم. در ضمن از آن‌جایی که if ای که قبلا نوشته بودیم بی‌معنا بوده، آن if را حذف می‌کنیم، ولی نکته بسیار مهم آن است که به هر حال چون با بانک اطلاعاتی کار می‌کنیم و امکان بروز خطا بالقوه وجود دارد، کماکان احمال خطای 500 را می‌دهیم و لذا آن را هم بالای Action می‌نویسیم!